Available *online* at : http://bit.ly/InfoTekJar

InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan

ISSN (Print) 2540-7597 | ISSN (Online) 2540-7600

Network Security

Analisis Forensik Serangan SQL Injection dan DoS Menggunakan Instrution Detection System Pada Server Berbasis Lokal

Dian kurnia

Universitas Pembangunan Panca Budi, Jl. Jend. Gatot Subroto Km. 4,5 Sei Sikambin, 2012, Medan, Indonesia

KEYWORDS

SQL Injection, DoS, Server, Forensic, IDS

CORRESPONDENCE

Phone: 082363333031

E-mail: diankurnia68@dosen.pancabudi.ac.id

A B S T R A C T

Analisa forensik adalah tindakan yang harus dilakukan *administrator network* dalam mengetahui sumber serangan yang terjadi pada server. Tindakan preventif perlu dilakukan bila diketahui banyaknya serangan yang *high priority* dari ancaman pada suatu server. Pada penelitian dilakukan skenario penyerangan pada suatu server yang dirancang, serangan yang dilakukan berupa serangan dengan teknik SQL injection. Teknik SQL injection yang diimplementasikan menggunakan teknik klasik yaitu dengan metode *comment attack* dan metode *numeric SQL Injection*. Pada Server juga telah disetting *snort* sebagai Intrution Detection System bila terjadi serangan maka fungsi snort akan bekerja dalam mendeteksi serangan SQL Injection. Dalam implementasi penelitian ini serangan SQL injection dengan metode *comment attack* mampu mem*bypass* login, dan menjadikan attacker login menjadi valid sebagai admin web. SQL injection dengan metode *numeric* dengan bantuan software sqlmap, kurang mampu memvalidasi user anymous menjadi admin, akan tetapi hal ini terdeteksi oleh system *snort* yang aktif. Pada serangan DoS, dilakukan sekenario serangan dengan fokus pada protokol TCP dengan port 22 ssh, data flooding yang besar memaksa snort untuk mendeteksi dan merekam ke *log* dengan cepat dan mngetahui keseluruhan IP address sumber serangan.

INTRODUCTION

Analisis forensic merupakan suatu tindakan administrator dalam menganalisis kejadian-kejadian (*incident*) yang terjadi pada suatu server. Kejadian dapat diketahui dengan melakukan penelusuran di beberapa log seperti log database server, log antivirus server, log aplikasi dari suatu server[1]. Ancaman yang sering terjadi pada suatu website yaitu penyerang memanfaatkan kelemahan dari database backend, yang mana penyerang menginjeksikan dengan melakukan komunikasi langsung pada SQLServer tanpa terindentifikasi valid atau tidak user yang mengakses tersebut pada suatu database. Hal ini dikenal juga dengan teknik SQL Injection. SQL injecton pada dasarnya mempunyai beberapa type antara lain SQL injection vulnerability Dengan SQL injection Attack[2]. SQL injection dapat dijuga dikategorikan dengan SQL injection klasik dan SQL injecton modern, hal ini berkembang sesuai dengan perkembangan keamanan query yang ada setiap development penyimpanan database seperti MAMP dan XAMP[3]. Selain serangan SQL injecton pada server, ada juga jenis teknik serangan yang dapat melumpuhkan kinerja server pada jaringan. Dengan cara melakukan pengiriman paket data secara menyeluruh ke port yang menjadi target dengan beberapa protocol pada jaringan seperti protocol TCP, UDP, dan HTTP. Dengan melakukan *flooding* pada server akan membuat server menjadi sibuk dan mudah mengalami down. jika hal ini terjadi maka server tidak dapat diakses oleh client dan kelemahan dari teknik *flooding* ini yaitu dalam melakuakan serangan membutuhkan bandwidth yang cukup besar dan lancar dalam melumpuhkan server target, akan tetapi pencegahan teknik sangat mudah yaitu dengan memblokir jalur ICMP agar setiap *attacker* tidak dapat mengirimkan paket ping yang berlebihan ke suatu server[4].

Penelitian Sebelumnya melakukan study untuk mengetahui perbandingan jenis teknik SQL Injection, perbandingan dilakukan dengan mengklasifikasikan SQLIA berdasarkan vulnerability atau dalam studynya disebut pendekatan statis, dengan pendekatan *dynamic*, dan juga pendekatan *hybrid*, hasil penelitian diketahui penangulangan pencegahan dari serangan SQL injection dengan beberapa type pendekatan, dengan adanya kombinasi antara pencegahan dan informasi teknik penyerangan amak membuat database suatu server menjadi aman[5].

Penelitian yang lainnya melakukan study mengenai data yang di rilis OWASP dimana serangan yang sangat rentan untuk suatu website adalah SQL Injectionyang mana teknik ini dapat mengganti informasi yang ada di dalam database. Beberapa

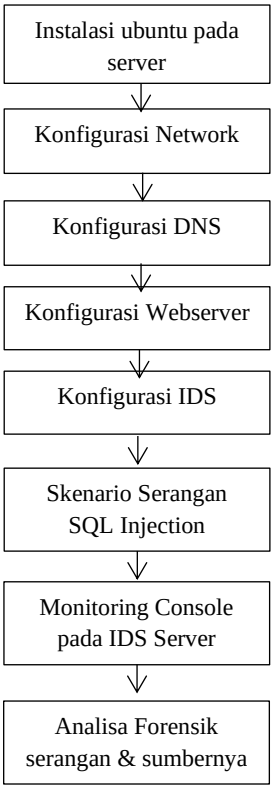
https://doi.org/10.30743/infotekjar.v4i2.2420

teknik vulnerability SQL injection juga di bahas dalam penelitian tersebut salah satu penulis tersebut melakukan teknik SQL Injection di dalam informasi bank. Sehingga diketahui teknik teknik SQL injection yang dapat melakukan injeksi dengan mudah pada suatu database hal ini menjadi sangat mungkin untuk melakukan pencegahan yang mana mekanisme dari serangan SQL injection tersebut telah diketahui bentuk bentuk implementasi[6].

Penelitian sekarang melakukan scenario serangan pada suatu server yang telah di setting sesuatu rancangan topologi jaringan. Serangan yang dilakukan yaitu *SQL Injection* dan *Deniel of Service*. Pada *SQL injection*, peneliti berfokus pada SQL injection dengan metode *comment attack* dan metode *numeric SQL Injection*. Pada sistem server telah dikonfigurasikan snort untuk mengidentifikasi serangan-serangan yang terjadi pada server. Data yang direkam di log snort dari aktivitas yang terjadi pada server akan dibandingkan dengan tabel klasifikasi untuk mengetahui deskripsi dari serangan yang terjadi, sehingga dapat juga mengetahui dampak priority dari serangan yang terjadi[7].

METHOD

Pada penelitian ini, penulis melakukan langkah-langkah instalasi dan konfigurasi untuk mendapatkan analisa forensik pada scenario jaringan yang di rancang. Adapun alur kerja penelitian dapat di lihat pada Gambar 1 berikut :



Gambar 1. Tahapan alur kerja

Pada Gambar 1 diketahui tahapan kerja berupa melakukan instalasi system operasi linux Ubuntu pada server, untuk hostnya dilakukan penginstalan juga linux Ubuntu versi CLI, Ubuntu server yang digunakan dalam penelitian ini yaitu Ubuntu server 12.0.4 LTS. Tahap selanjutnya dilakukan konfigurasi network dengan mensetting interface pada server dan host. Pada server dilakukan konfigurasi DNS Server, dimana dalam penelitian digunakan nama DNS Server yaitu www.unpab.net , webserver yang dikonfigurasi menggunakan database Mysql Server dengan phpmyadmin. Adapun skenario jaringan yang di rancang dapat di lihat pada gambar 2 berikut.

Gambar 2. Skenario jaringan sebelum Attacker

Pada gambar 2 diatas di ketahui bahwa sumber internet dari modem diteruskan ke suatu server yang sudah dikonfigurasi Webserver dan DNS pada skenario ini belum terkonfigurasi *Intrution Detection System*. Apabila terjadi serangan pada kondisi ini maka server dan administrator tidak dapat mengidentifikasi serangan yang terjadi pada server itu sendiri. Perlu adanya skenario firewall tambahan, seperti IDS dan pemblokiran ICMP untuk mendeteksi skenario serangan yang bila terjadi pada suatu server.

Gambar 3. Skenario jaringan setelah Attacker

Pada tahap inilah dikonfigurasi *Intrution Detection System*, yang mana skenario serangan terjadi pada port eth0 pada server. Pada network eth0 dikonfigurasi ip address dengan mode *inet dhcp* dan pada eth1 dikonfigurasi dengan mode ip *inet static*. Untuk melakukan pengauloadan pada file pada webserver menggunakan aplikasi Winscp dan mode SFTP dengan username : unpab dan password 1. IP DNS akses yaitu 192.168.43.114, sehingga untuk menjalankan phpmyadmin pada browser mozilla ataupun firefox maka digunakan ip address 192.168.43.114/phpmyadmin, dengan username : root password : 1. Pada server di konfigurasi juga *intrution detection system*. Settingan rules pada IDS ini difokuskan untuk memonitoring port eth0. Pada tahap perancangan ini, penulis hanya melakukan monitoring pada console terminal server tidak melakukan *prevetion* / pencegahan, rules snort dapat di lihat pada `/etc/snort/rules/local.rules.[8]`

Skenario penyerangan yang dirancang oleh penulis dimana teknik penyerangan yang dilakukan yaitu *MYSql Injection* dan *Denial of Services* (DoS). *Mysql injection* akan difokuskan dalam menyerang webserver yang telah di bangun. Akan dibuat page login sederhana yang mempunyai database yang terintegrasi pada phpmyadmin server. Notifikasi error akan dimunculkan pada browser jika *username* dan *password* yang diinputkan oleh *user* ataupun *hacker* tidak sesuai pada form proses cek login. Teknik Mysql yang akan dilakukan menggunakan *methode comment attack*.

Jenis serangan ini memanfaatkan komentar inline diizinkan oleh SQL - kode berbahaya dan komentar apa pun yang muncul setelah "-" dalam klausa WHERE. Itu intinya adalah bahwa semuanya setelah karakter komentar akan diabaikan. Komentar Attack dapat dikombinasikan dengan keduanya String atau Numeric SQL Injection sehingga berfungsi sebagai tautologi yang selalu mengevaluasi pernyataan yang benar[2].

Skenario :

```
User Input: ‘’-‘ OR ‘1’ =’1 –‘.
Generated SQL Query: SELECT username, password
FROM user WHERE username = ‘’- OR ‘1’ =’1 –‘ AND
password = ‘,’
```

dan melakukan *tools* SQLmaps untuk melakukan numeric SQL Injection yang *compatibility* dengan ubuntu versi CLI untuk *client*[2].

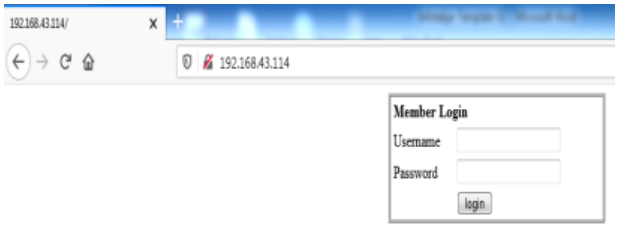
Skenario :

```
Normal Statement: SELECT * FROM user  WHERE id=
'2'
Input: 1 Output: id 2's Rows only.
Injected Statement: SELECT * FROM user WHERE
name= '2' OR '1' ='1'.
Input: '2' OR '1' ='1'
Output: this will return rows for'2'id or
wherever one equals to one (ALL ROWS)
```

Diharapkan data teknik *SQL injection* akan diterapkan mampu running normal sehingga *intrution detection system* dapat mengidentifikasi serangan yang terjadi pada server. Serangan tahap kedua penulis melakukan teknik *Denial of Services* (DoS) menggunakan sistem operasi linux ubuntu juga. Dalam hal penulis memanfaatkan software DDoS Attack dengan target IP address yang tertuju pada server menyimpan data webserver yang telah di rancang. Serangan *Denial of Services* (DoS) difokuskan pada protokol TCP dan UDP dengan port pada serangan di fokuskan juga pada port 22 yaitu SSH pada server tersebut. Adanya serangan *Denial of Services* (DoS) diharapkan server yang dirancang dan telah memiliki *intrution detection system* dapat mampu mengidentifikasi serangan DOS yang terjadi. Rancangan IDS ini juga diharapkan dapat mengetahui sumber serangan dari hacker dimana IP Address hacker dapat direkam pada log system dari snort terdapat pada *intrution detection system* Server.

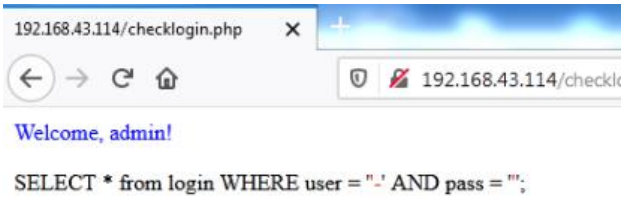
RESULTS AND DISCUSSION

Dalam mencapai hasil penelitihan maka diperlukan implementasi sesaui dengan rancangan skenario jaringan dan metode penyelesaian. Dalam implementai penyerangan penulis melakukan dengan teknik MYSQL Injection dan teknik DoS. Adapun teknik MySql injection yang dilakukan penulis dapat di lihat pada Gambar 4 berikut :



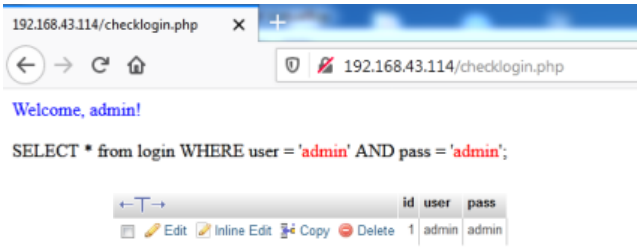
Gambar 4. Pengujian akses webserver

Pada Gambar 4, pengujian akses webserver dengan computer hacker, ip webserver 192.168.43.114 dengan nama domain : www.unpab.net . Tampilan default username dan password pada page login sebelum terjadi penyerangan dengan MYSql Injection. Tahap teknik SQL Injection dengan melakukan penginputan karakter khusus dalam hal ini penulis melakukan penginputan karakter pada username = ‘- dan password = , kemudan ketika di klik login maka hacker dapat mudah masuk login pada suatu website. Adapun tampilan dapat di lihat pada Gambar 5 berikut :



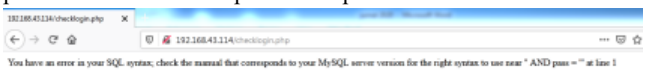
Gambar 5 Hacker berhasil login dengan penginputan karakter khusus

Hacker tersebut berhasil masuk kedalam website dengan teknik MYSQL Injcetion, hal ini diketahui *username* dan *password* yang terdapat pada database Mysql Server dengan id=1, username = admin & password = admin. Adapun dapat di lihat pada Gambar 6 berikut :



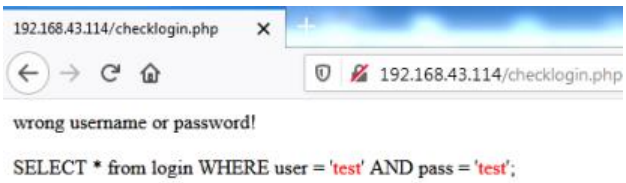
Gambar 6. Isi tabel user dan pengujian autentikasi dengan username = admin dan password = admin

Pada dasarnya sebelum melakukan serangan dengan teknik SQL injection maka perlu teknik penginputan karakter khusus agar bug kesalahan SQL Injection di ketahui, pada tahap ini sebelum penulis melakukan uji coba pengiputan username = test’- dan password = ‘. Hal ini dapat di lihat pada Gambar 7 berikut :



Gambar 7 Penginputan karakter khusus untuk mengetahui bug

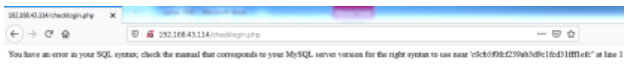
Ketika dilakukan pengujian dengan username yang tidak sesaui dengan database maka diketahu hasil ceklogin.php dengan aller = wrong username or password!. Dapat di lihat pada Gambar 8 :



Gambar 8. *Username* dan *password* tidak sesuai database

https://doi.org/10.30743/infotekjar.v4i2.2420

Tahap selanjutnya variabel username dan password telah diamankan dengan algoritma MD5, kemudian dilakukan teknik penyerangan dengan username = '- dan password = ,
maka login yang sebelumnya dapat masuk oleh hacker dengan menggunakan algoritma MD5 maka attacker tidak dapat masuk kedalam website yang telah di rancang.



Gambar 9 Teknik SQL Injection pada website yang telah di amakan algoritma MD5

```
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'c0cb5f0fcf239ab3d9c1fcd31fff1efc'' at line 1
```

Percobaan selanjutnya pengujian menggunakan username = admin dan password = admin, maka akan diketahui pass yang benar dan terenkripsi algoritma MD5. Dapat di lihat pada Gambar 10.

Gambar 10 Login website dengan username = admin dan password = admin

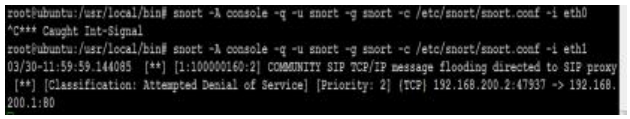
Untuk mengetahui deskripsi dari
pass = `21232f297a57a5a743894a0e4a801fc3` maka perlu pengujian pada url berikut :
https://www.md5online.org/md5-decrypt.html

Gambar 11 MD5 Deskripsi dari variable pass

Teknik serangan SQL Injection dengan menggunakan tools sqlmap menggunakan perintah :
./sqlmap.py –u http://www.unpab.net.index.php?id=1 –dbs

Gambar 12 Serangan SQL Inection

Hasil serangan pada Gambar 12 diketahui sqlmap tidak mendeteksi adanya bug kesalahan pada website, oleh karena itu hasil berupa tampilan database yang digunakan oleh administrator web tidak dapat diketahui. Sehingga hasil monitoring pada *console* di terminal linux Ubuntu server, ketika di monitoring pada port eth0 maka tidak terdeteksi pasti bahwa serangan tersebut adalah SQL Injection melainkan serangan DDoS, hal ini dikarenakan sqlmap dari attacker tidak dapat menemukan bug untuk masuk menscan datase yang digunakan pada server.

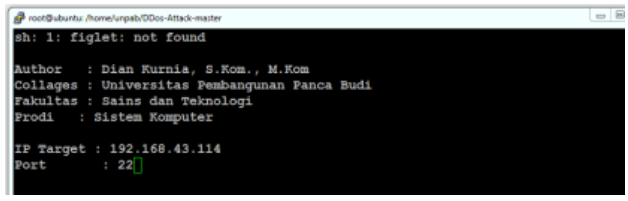


Gambar 13 IDS menotifikasi adanya serangan SQL Injection

Pada serangan SQL injection pada Gambar 13, serangan bersumber dari IP address 192.168.200.2 dengan port : 47937, serangan ini terbaca pada system yaitu *Attempted Denial of Service* (priority 2) yaitu kita cek pada table 1 diketahui kelas type dari serangan ini termasuk category medium, dimana efek serangan secara tidak langsung mempengaruhi kinerja dari CPU server jika dibiarkan oleh administrator.

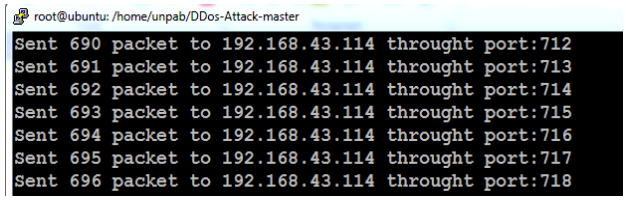
Teknik kedua scenario penyerangan yaitu menggunakan Teknik *Denial of Service*, dimana teknik biasanya dilakukan oleh seorang attacker bertujuan untuk melakukan flooding data yang berlebih/pengiriman paket data yang overload, sehingga kapasitas/kemampuan server tidak dapat melayani permintaan data dari computer host attacker. Hal ini bisa saja mengancam kelancaran jalur jaringan pengaksesan server oleh client. Scenario pada serangan ini, pertama penulis menggunakan tools DDoS Attack dengan bahasa pemrograman python. Perintah syntax yang digunakan :

```
root@ubuntu:/home/unpab/DDos-Attack-master# python ddos-attack.py
```



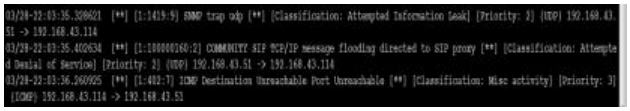
Gambar 14 pengujian Denial of Services

Pengujian DoS dilakukan dengan menginputkan ip target pada hal ini ip addres server pada eth0 yaitu 192.168.43.114, kemudian attacker berfokus pada port ssh untuk menghentikan jalur remote server yang digunakan oleh administrator network. Hasil selanjutnya mulailah teknik ini mengirimkan paket dengan sejumlah n+1 dengan periode waktu detik (s), dan tidak berfokus kali dengan port 22 melainkan mengirimkan paket yang lama kelamaan bernilai besar ke seluruh port yang kemungkinan terbuka pada server tersebut. dapat di lihat pada Gambar 15 berikut :



Gambar 15 Pengiriman paket dengan teknik DoS

Hasil pengujian pengiriman paket data yang dikirimkan melalui teknik DoS terindentifikasi oleh snort pada port eth0 console terminal linux Ubuntu. Dapat di lihat pada Gambar 16 berikut :



Gambar 16 Hasil snort ketika teknik DoS berjalan

Dari hasil analisa forensic dengan snort console serangan DoS berjalan pada protocol TCP[9] dapat di lihat pada Gambar 16 divdapat hasil diketahui serangan dengan class type *Attampted Information Leak* dengan *priority medium*, serangan bersumber dari IP = 192.168.43.51 kemudian diidentifikasi serangan juga dengan class type = Attempt Denialof Service dengan priority = medium pada port TCP/IP karena pada serangan kita fokus menyerang jalur protokol ini. Kemudian Classfication Misc Activity merupakan priority medium yang semua serangan tersebut bersumber dari IP=192.168.43.51, penanganan pada tingkat priority medium harus juga dilakukan dalam kasus DoS, pencegahan dapat dilakukan dengan memblokir akses IP pada computer host attacker, ataupun mendapatkan informasi mengenai MAC address pada computer attacker tersebut hal ini bisa dilakukan secara menual jika computer attacker hanya 1 host saja. Keterangan dari serangan-serangan ini berdasakan pada tabel 1 berikut :

Table 1. *Snort Default Classifications*[10]

Class type	*Description*	*Priority*
web-application-attack	Web Application Attack	high
attempted-dos	Attempted Denial of Service	medium
attempted-recon	Attempted Information Leak	medium
bad-unknown	Potentially Bad Traffic	medium
denial-of-service	Detection of a Denial of Service Attack	medium
misc-attack	Misc Attack	medium
successful-dos	Denial of Service	medium
successful-recon-limited	Information Leak	medium
web-application-activity	Access to a potentially vulnerable web application	medium
icmp-event	Generic ICMP event	low
misc-activity	Misc activity	low
network-scan	Detection of a Network Scan	low
unknown	Unknown Traffic	low

CONCLUSIONS

Analisa *forensic* dalam penelitian ini diketahui sangatlah diperlukan bagi administrator *network*, yang mana teknik SQL injection tidaklah mudah diterapkan oleh seorang *attacker* apabila *bug* dari suatu website tidak ditemukan dan adanya juga pengamanan enkripsi pada page login password MD5 enkripsi, dikarenakan untuk mengetahui *bug* suatu website dibutuhkan kemampuan yang cukup mengenai *structure* dari MYSql, dalam artian SQL injecton akan terjadi jika memang ada kesalahan script pada website, hasil deteksi snort yang ditemukan dari percobaan *attacker* menggunakan software sqlmap tapi hasil scannya tidak menemukan database maka serangan terdeteksi oleh snort sebagai serangan DDoS dengan tingkat priority 2 (medium), Pada pengujian teknik serangan DoS yang dilakukan dengan berfokus pada port 22 ssh, hasil yang didapat teknik ini mampu menyerang fokus pada protocol TCP/IP pada eth0 server. Hal ini juga membuat snort harus lagi mendeteksi adanya serangan dari DoS dengan hasil *flooding attack*, yang sebelumnya terdeteksi *Attempted Information Leak*, kemudian jalur ICMP pun menjadi *uncreable* dikarenakan serangan DoS.

ACKNOWLEDGMENT

Dalam penelitian ini juga konstribusi dari community yang mengembangkan script python DDoS Attack dan beberapa community project sqlmap untuk dapat support pada Ubuntu 12.2.04 yang tersimpan para repository website github. Beberapa referensi dari peneliti-peneliti sebelumnya mengenaia SQL injection yang bersumber dari jurnal nasional dan international journal.

REFERENCES

[1] S. Universitas, G. Mada, and G. Mada, “Analisis Forensik Jaringan Studi Kasus Serangan SQL Injection pada Server Universitas Gadjah Mada,” vol. 6, no. 2, 2012.

[2] D. A. Kindy and A. K. Pathan, “A Detailed Survey on Various Aspects of SQL Injection in Web Applications : Vulnerabilities , Innovative Attacks , and Remedies,” pp. 1–13, 2012.

[3] Z. S. Alwan and M. F. Younis, “Detection and Prevention of SQL Injection Attack : A Survey,” vol. 6, no. 8, pp. 5–17, 2017.

[4] D. Kurnia, “Analisis Pertahanan Website pada Protokol TCP dan UDP dari Serangan DDoS,” no. x, 2018.

[5] S. Mohammad, S. Sajjadi, and B. T. Pour, “Study of SQL Injection Attacks and Countermeasures,” vol. 2, no. 5, 2013.

[6] V. Syamasudha, A. R. Syed, and E. Gayatri, “The Solutions of SQL Injection Vulnerability in Web Application Security,” no. 6, pp. 3803–3808, 2019.

[7] D. Kurnia, U. Pembangunan, P. Budi, J. Jend, and G. Subroto, “ANALISIS KEAMANAN SERVER MENGGUNAKAN SNORT DARI SERANGAN PORT SCANNING DAN DOS,” 2019.

[8] N. Dietrich, “This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0),” 2015.

[9] N. Forensics, *No Title*. .

[10] “2.2 Preprocessors - Snort manual.” [Online]. Available: http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node17.html. [Accessed: 01-Apr-2020].

AUTHOR(S) BIOGRAPHY

Dian Kurnia
Dosen di Universitas Pembangunan Panca Budi Medan, pada Fakultas Sains & Teknologi di Program Studi Sistem Komputer konsentrasi Keamanan Jaringan Komputer, mulai bekerja sebagai dosen mulai Desember 2016, fokus bidang penelitian pada bidang Security System, Analyst Network dan Administration network, cloud data mining. S2 Teknik Informatika Universitas Sumatera Utara dan S1 Teknik Informatika STMIK Budidarma.

 https://doi.org/10.30743/infotekjar.v4i2.2420